Domain Ontology for Personalized E-Learning in Educational Systems

José M. Gascueña, Antonio Fernández-Caballero & Pascual González
Laboratory of User Interaction and Software Engineering (LoUISE)
Computer Science Research Institute
University of Castilla-La Mancha, Albacete, Spain
{jmanuel, caballer, pgonzalez}@info-ab.uclm.es

Abstract

This paper introduces a domain ontology to describe learning material that compose a course, capable of providing adaptive e-learning environments and reusable educational resources. Two characteristics have been considered to describe each resource: (1) the most appropriate learning style and, (2) the most satisfactory hardware and software features of the used device. Basically, we have adopted the Felder-Silverman Learning Style Model for the learning styles and we have based in the FIPA Device Ontology for the description of the devices. Also, some elements from the IEEE LOM Standard have been chosen to describe other metadata of the learning resources. The ontology has been developed under OWL language, the last standard language by the W3C to represent ontologies in the Web.

1. Introduction

Unfortunately, current e-learning platforms do generally not include or even consider some important characteristics capable of providing user adaptivity in a satisfactory manner. For instance, they do not pay attention on the students' learning styles; thus, all students are shown the same materials and activities. Nor do didactic materials offer any reusability possibility due to the lack of granularity or access possibilities to different devices (PC, PDA, cell phone, and so on) in an efficient way. Ontologies are a promising research domain to overcome the most common problems for intelligent educational applications [1], [2]. Ontologies allow to specify formally and explicitly the concepts that appear in a concrete domain, their properties and their relationships. Furthermore, they are useful in many environments: and especially in educational environments, as they enable people and/or software agents to share a common understanding of the knowledge structure. Moreover, they permit to reuse knowledge, that is to say, it is not necessary to develop an ontology from scratch if another ontology is available for use in the modeling of the current domain.

This paper addresses the three issues commented by using a domain ontology in OWL language [3] - the last standard language proposed by the W3C to represent ontologies in the Web – with the objective of reflecting the structure of educational contents. The Protégé 3.1.1 framework [4] has been selected to edit/construct the contents. In the proposed ontology the fundamental components are the learning objects. To us, a learning object is anything digital that can be delivered across the network on demand, be it large or small (text, images, audio, video, animations, applets, entire web pages that combine several media types, and so on). In the description of a learning object, there are two important characteristics included to get adaptation: (1) the appropriate learning style and, (2) the features of the device to display learning objects correctly.

To describe the learning objects we will use metadata. In this work, we have chosen some elements of the IEEE LOM, an internationally recognized and adopted standard [5]. On the other hand, in order to describe the device characteristics, we have based in the elements of the technical category of the LOM and on the FIPA Device Ontology [6]. FIPA Device Ontology specifies a frame-based structure to describe devices, and it is intended to facilitate agent communication for purposes such as content adaptation though terminal devices as PC's, PDA's and the like. Lastly, the Felder-Silverman Learning Style Model (FSLSM) [7] has been selected to describe the learning style that best fits an object, as LOM does not manage this issue.

In the same direction, some proposals to organize learning objects in courses have been introduced so far. Ronchetti and Saini [8] define an ontology to describe the contents of the e-learning material Computer Science, based in ACM Computing Curricula 2001, and propose an architecture to aid students in finding materials that present different points of view or different ways to explain concepts. On the other hand, in [9] we find another approximation to the Computing Curricula 2001, which gives an example of an ontology developed to teach only a concrete subject, namely the Java language. Silva describes ontology-based metadata to achieve personalization and reuse of content in the AdaptWeb project [10], where DAML+OIL language is used to represent the ontology. In [11] the authors propose to use several RDF ontologies for building adaptive educational hypermedia systems. These ontologies describe the features of the domain and of the learner, as well as observations about the learner's interactions with the e-learning system, and of the presentation for generating hypertext structures. Generally, learning styles are not considered in the previous works (except in [10]). Amorim et al. [12] base their proposal in the FIPA Device Ontology to describe the devices used in the EUME project (PDA, desktop computer, video projector, video camera, and so on).

The rest of this paper is organized as follows. In section 2 our complete ontology is introduced. Conclusions and future work are described in section 3.

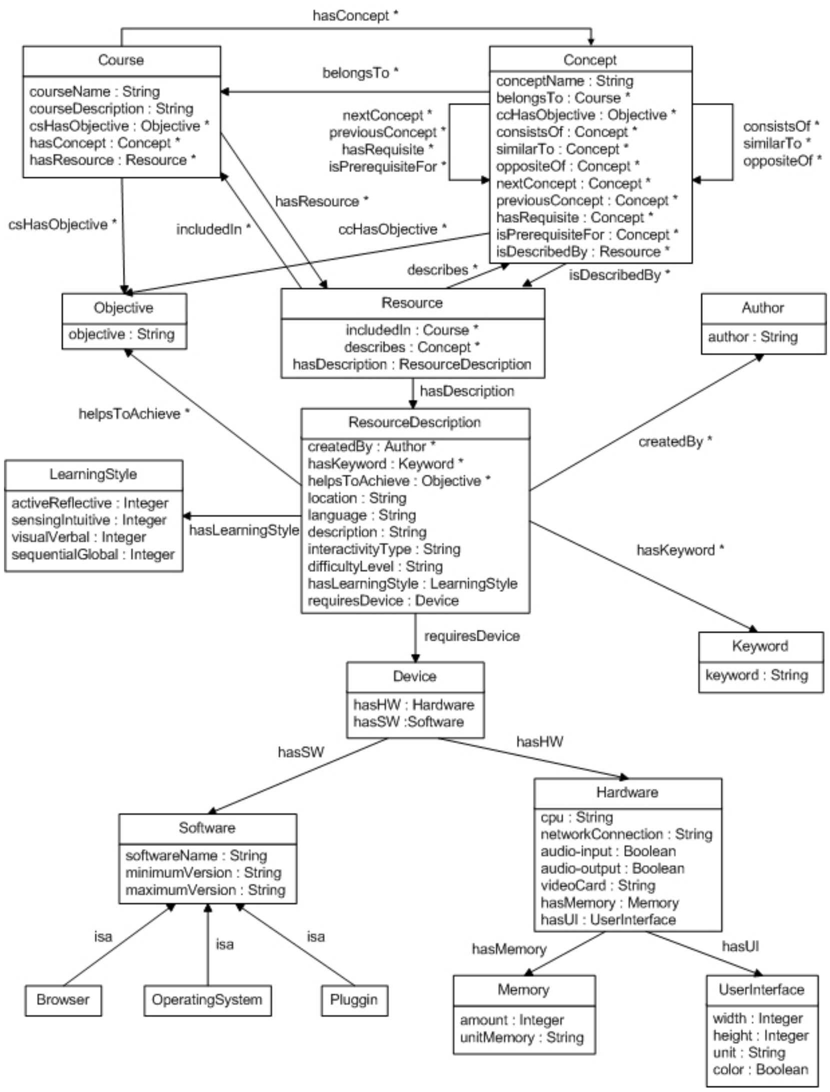

Figure 1: General layout of the domain ontology

2. Description of the ontology

Figure 1 shows the whole layout of the domain ontology proposed in this paper.

Firstly, let us start with class *Course*, which represents the subjects being taught in an educational application. For example, *Multiagent Systems* could be an individual of this class. The class contains several properties: *courseName* and *courseDescription* are the name and a brief description of the course, *csHasObjective* points to the objectives to reach (class *Objective*), whereas *hasConcept* (*belongsTo* is its inverse) and *hasResource* point to the set of concepts and resources, respectively, that compose a course.

The concepts constitute the knowledge of the treated domain and they are collected in class *Concept*. This class contains data type property *conceptName*, to identify the concept, and other object properties that allow to establish different relations among domain concepts: (a) *consistOf* serves to define a concept hierarchy, and therefore, to establish a relation among a concept and its sub-concepts (e.g. we are able to define chapters, sections, subsections and terms which are under sections), until reaching an atomic concept which - from the point the view of the teacher – does not need to be decomposed any more, (b) *similarTo* and *oppositeOf* make it possible to map a concept to other concepts that have the same or different semantic meaning, respectively, (c) *nextConcept* and *peviousConcept* indicate the concepts through which it is possible to advance/go back from a given concept – the browsing possibilities reflected with these two properties do not impose any constraint on the mapped concepts to be known or not, and (d) *hasRequisite* and *isPrerequisiteFor* (its inverse) allow to point to concepts that must be known before starting to study a concept, and the concepts for which it is a prerequisite, respectively. In this case, some conditions should be fulfilled to accede to the study of the concepts. On the other hand, with the study of a concept, a collection of objectives pointed by the object property *ccHasObjective* is achieved.

The object property *isDescribesBy* (class *Concept*) points to digital resources that explain a concept or assess the knowledge stored about it. The capacity of obtaining a high grade of reusability for a learning object is largely a function of the granularity of the objects. We consider the learning objects that have a very low granularity as resources, that is to say, at the level of paragraph, image, table, diagram, and so on. Thus, in every moment an e-learning system is able to add/remove contents at this level and to produce tailor-made learning materials according to the preferences of a student. Also, this facilitates showing didactical materials in those devices that have a screen of limited dimensions (PDA) in form of a sequence of pages. Learning objects that have a bigger granularity are built from smaller granularity ones. For instance, the course chapter's section will be created by mixing several little chunks (this is the way we have named the resources), the sections will form the chapters and the latter a course. Thus, by means of this process we are able to reuse learning objects at different levels.

A resource can be included in several courses (object property *includedIn* – *hasResource* is its inverse), and it can reference several concepts (object property *describes* – *isDescribedBy* is its inverse). Moreover, class *Resource* includes an object property *hasDescription* to point to the description of class *ResourceDescription*, where more metadata that describe a resource are described. As you may observe, a resource (a) is created by one or several authors (property *createdBy*), (b) has a set of keywords that describe it (property *hasKeyword*), (c) helps to reaching a few objectives (property *helpsToAchieve*), (d) is located in a certain direction (property *location*), (e) is written in a given language (property *language*), (f) has a brief description (property *description*), (g) incorporates a type of interactivity - it can take values active, exhibition and mixed - (property *interactivityType*), and, (h) possesses a grade of difficulty - very easy, easy, average, difficult, and very difficult - (property *difficultyLevel*). The type active applies for documents where the student interacts and/or performs operations (for example, simulations, exercises, test questionnaires), whereas exhibition is applied to documents whose objective is that the student gets the content (for example, text, images, sound). Lastly, in order to point to the learning styles that are better adjusted to a resource and that are more correctly visualized on a device, object properties *hasLearningStyle* and *requiresDevice*, respectively, are introduced.

2.1. Learning styles description

We suppose that the scheme to distinguish the student's learning style is the one proposed by the Felder-Silverman Learning Style Model (FSLSM) [7]. We have adopted this supposition for two reasons. First of all, this model provides a questionnaire to establish the dominant learning style of each student [13] and its results can be linked easily to e-learning systems. Second, this model is sufficiently validated in many adaptive environments [14], [15], [16], [17]. Therefore, if we have classified all the learning objects using this model, it is possible to deliver contents adapted to student's learning styles. Class *LearningStyle* of the ontology represents the learning styles that the learning objects are able to include. This class offers four properties of type integer that correspond to the dimensions of the FSLSM (active-reflective, visual-verbal, sensing-intuitive, sequential-global).

2.2. Device description

Class *Device* describes the necessary technology to use a resource. In order to visualize the learning objects correctly and in a suitable time, it is necessary to have a device that satisfies certain hardware (class *Hardware*) and software (class *Software*) requirements. With regard to hardware, we consider features such as the computer CPU type (*cpu*), the network connection required (*networkConnection*), the necessary memory (*object property hasMemory*), or the user interface characteristics (object property *hasUI*). We are also interested in knowing if it is necessary to have the capability to receive audio input (*audio-input*) or to produce audio output (*audio-output*), as well as information on the video card (*VideoCard*). Property *hasMemory* points to requirements that the memory should have - the amount of memory necessary to show a resource to a user (*amount*) and the unit used to express it (*unitMemory*); whereas property *hasUI* indicates the information that describes the user interface - the width of the screen (*width*), the height of the screen (*height*), unit for the width and the height parameters (*unit*) and if a color screen is needed (*color*). Regarding the software, we include features such as the minimum (*minimumVersion*) and the maximum (*maximumVersion*) version capable of using the resource and the name that identifies it (*softwareName*). Also, we distinguish among several software types: browser (class *Browser*), operating system (class *OperatingSystem*), and pluggins (class *Pluggin*).

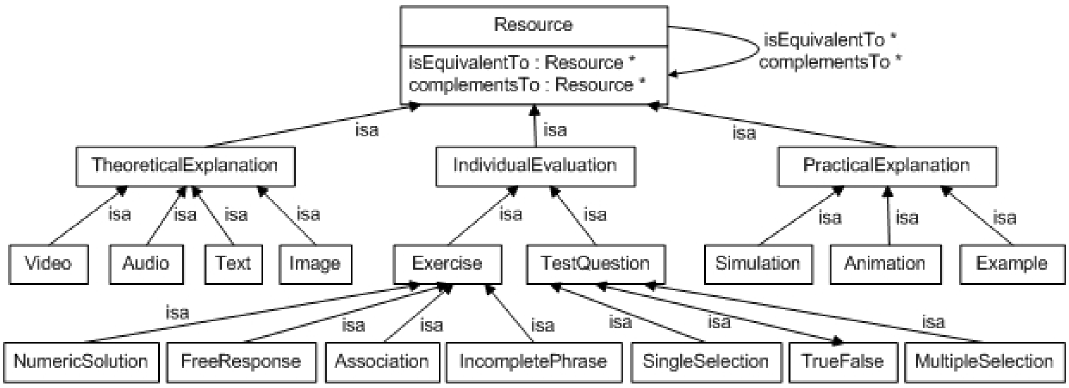

Figure 2: The Resource class

2.3. Resource class

Class *Resource* is divides into three subclasses: *TheoreticalExplanation*, *PracticalExplanation* and *IndividualEvaluation*. Classes *TheoreticalExplanation* and *PracticalExplanation* represent the theoretical and practical explanations, respectively, displayed to the students (see Figure 2). This distinction allows showing to the sensory students first the practical applications of the theory and later the purely theoretical contents - and vice versa for the intuitive students. To compose the theoretical explanations, several types of formats are proposed there (classes *Text*, *Audio*, *Video*, and *Image*). This way, the theoretical explanations that appear to the verbal students are formed by text and/or audio, whereas videos and images are shown to the visual students. In order to realize practical explanations, examples, simulations and animations (classes *Example*, *Simulation*, *Animation*) can be used. The individuals of class *IndividualEvaluation* are used to evaluate the knowledge acquired by the student. Class *IndividualEvaluation* has two subclasses (*Exercise* and *TestQuestion*) that contain the exercises and the test questionnaires, respectively, that a student has to solve. Class *Exercise* contains four subclasses that correspond to statements in which we have to answer with a number (*NumericSolution*), to complete one or more points with a phrase, a word or a cipher (*IncompletePhrase*), to answer with one or several paragraphs to a question (*FreeResponse*), or to establish relations among elements that appear in two parallels columns (*Association*). On the other hand, class *TestQuestion* incorporates several subclasses to highlight the different types of test questions that are shown to the student. For instance, the student has to choose among one of two alternatives (*TrueFalse*), one of three or more alternatives (*SingleSelection*), all the correct alternatives (*MultipleSelection*). To conclude, class *Resource* also includes the *object properties isEquivalentTo* and *complementsTo*. The fist one is to know the resources that have the same semantic meaning, whereas the second allows reusing resources that the author of the contents thinks are necessary to group together in a course. For instance, a diagram should be grouped together with the paragraph and/or audio that describes it; a simulation should be reused together with the paragraphs that explain a concept, etc.

4. Conclusions and future work

In this paper we have proposed a domain ontology to describe learning materials that compose an adaptive course. We have considered two interesting aspects; namely, the learning style more adequate for an educational resource, and the device that best uses it. The inclusion of the learning style in the description of the learning objects enables that an e-learning system compares the user's learning style with the resources' learning style possibilities. This way, the learning materials that best fit the student's individual requisites are dispatched. Moreover, we have justified the benefits of considering as learning objects those resources that possess a fine-grain granularity (paragraph, image, diagram, and so on) in order to achieve a good reuse of the educational resources and a good visualization in devices with little windows.

As a future work, we aim to create a tool that enables to catch the courses taught in our University from the ontology proposed. This should not represent too much work, as we have developed a distance teaching/learning system so far [18], where, in order to obtain good results, the matter to be taught is decomposed into theory, exercises and test questionnaires. The current system should be greatly enhanced in its adaptation capabilities by incorporating the ontology proposed in this paper.

5. References

[1] Mizoguchi, R. & Bourdeau, J. (2000) "Using Ontological Engineering to Overcome Common AI-ED Problems", International Journal of Artificial Intelligence in Education, vol. 11, 2000, pp. 1-12.

[2] Devedzic, V. (2003) "Key issues in next-generation web-based education", IEEE Transactions on Systems, Man, and Cybernetics, Part C, vol.33, no.3, pp. 339-349.

[3] World Wide Web Consortium. (2004). "OWL Web Ontology Language Overview".

[4] Protégé. http://protege.stanford.edu

[5] Draft Standard for Learning Object Metadata. http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

[6] FIPA Device Ontology Specification. 6-12-2002. http://www.fipa.org/specs/fipa00091/SI00091E.pdf

[7] Felder, R.M & Silverman, L.K. (1988). "Learning and teaching styles in engineering education". Journal of Engineering Education, vol. 78, no. 7, pp. 674-681.

[8] Ronchetti, M. & Saini, P. (2004) "Knowledge management in an e-learning system". Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies, pp. 365-369.

[9] Lee, M.C., Ye, D.Y. & Wang, T.I. (2005) "Java learning object ontology". Proceedings of the 5th IEEE International Conference on Advanced Learning Technologies, pp. 538-542.

[10] Silva Muñoz, L. (2004) "Ontology based Metadata for e-learning Content". Master Thesis. Federal University of Rio Grande do Sul, Porto Alegre, Brazil.

[11] Henze, N., Dolog, P. & Nejdl, W. (2004) "Reasoning and ontologies for personalized e-learning in the semantic web". Educational Technology & Society, vol. 7, no. 4, pp. 82-97.

[12] Amorim, R., Lama, M., Sánchez, E. & Vila, X.A. (2003) "An educational ontology based on metadata standards". 2nd European Conference on e-Learning, pp. 29-36.

[13] Soloman, B.A. & Folder, R.M. (2006). "Index of Learning Styles Questionnaire". http://www.engr.ncsu.edu/learningstyles/ilsweb.html

[14] Shi, H., Rodriguez, O., Shang, Y. & Chen, S. (2003) "Integrating adaptive and intelligent techniques into a web-based environment for active learning". Intelligent Systems: Technology and Applications, vol. 4, chapter 10, pp. 229-260.

[15] Hong H. & Kinshuk. (2004) "Adaptation to student learning styles in web based educational systems". Proceedings of ED-MEDIA 2004, World Conference on Educational Multimedia, Hypermedia & Telecommunications, pp. 491-496.

[16] Capuano, N., Gaeta, M., Micarelli, A. & Sangineto, E. (2005) "Automatic student personalization in preferred learning categories". Proceedings of the 3rd International Conference on Universal Access in Human Computer Interaction.

[17] Peña, C.I., Marzo, J.L. & De la Rosa, J.L. (2005) "Intelligent agents to improve adaptivity in a web-based

learning environment". Knowledge-Based Virtual Education, pp. 141-170.
[18] Fernández-Caballero, A., Gascueña, J.M., Botella, F. & Lazcorreta, E. (2005) "Distance learning by intelligent tutoring system. Part I: Agent-based architecture for user-centred adaptivity". Proceedings of the 7th International Conference on Enterprise Information Systems, vol. 5, pp. 75-82.